CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CE145 : Digital Electronics
AY : 2022-23

Chapter 2
Boolean Algebra (Part 1)

Presented By :
Prof. Bhavika Patel
DEPSTAR, FTE

DEPSTAR

Devang Patel Institute of Advance Technology and Research

Boolean Algebra may be defined with

➢ A set of elements

➢ A set of operators and

➢ A numbers of unproved postulates

Set of elements: A set of elements is any collection of objects having a common property.

$$A = \{1,2,3,4\}$$

• If S is a set, and X and Y are certain objects, then X ∈ S denotes X is an object of set S, whereas Y ∉ S denotes Y is not the object of set S.

Binary Operator: A Binary Operator defined on a set S of elements is a rule that assigns to each pair of elements from S and derive unique element from S.

i.e. Consider the relation, a * b = c

* is a binary operator, if it specifies a rule, for finding c from the pair (a,b), and also a,b,c∈S.

However, * is not a binary operator if a, b ∈S , while the rule finds c ∉ S

Postulates: Assumptions

1. **Closure:** A set is closed with respect to a binary operator if, for every pair of elements of S, the binary operator specifies a rule for obtaining a unique element of S.

   for the set of Natural Numbers N = {1,2,3,4,5.......}

   N is closed with binary operator (+) for a,b $\in$ N we obtain unique c $\in$ N

   But N is not closed with Binary operator (-)

   because $2 - 3 = -1$ when $-1 \notin$ N

2. Associative Law : A binary operator (*) on a set S is said to be associative whenever

$$(X * Y) * Z = X * (Y * Z) \quad \text{for all } X,Y,Z \in S$$

3. Commutative Law: A binary operator (*) on a set S is said to be commutative whenever

$$X * Y = Y * X \quad \text{for all } X,Y \in S$$

4. Identity Element: A set S is to have an identity element with respect to a binary operation (*) on S, if there exists an element $E \in S$ with the property

$$E * X = X * E = X \quad \text{for every } X \in S$$

i.e. 0 is identity element with respect to operation (+)

$$0 + X = X + 0 = X \quad \text{for every } X \in S$$

5. Inverse: If a set S has the identity element E with respect to a binary operator (*), there exists an element $X \in S$, which is called the inverse, for every $Y \in S$, such that

$$X * Y = E$$

i.e. In the set of integers I with $E = 0$, the inverse of an element X is (-X) since

$$X + (-X) = 0$$

6. Distributive Law: If (*) and (+) are two binary operators on a set S, (*) is said to be distributive over (+), whenever

$$A * (B + C) = (A * B) + (A * C)$$

DEFINITION OF BOOLEAN ALGEBRA

In 1854 George Boole introduced a systematic approach of logic and developed an algebraic system to treat the logic functions, which is now called **Boolean algebra**.

In 1938 C.E. Shannon developed a **two-valued Boolean algebra** called Switching algebra, and demonstrated that the properties of two-valued or bistable electrical switching circuits can be represented by this algebra.

TWO-VALUED BOOLEAN ALGEBRA

Two-valued Boolean algebra is defined on a set of only two elements, **S={0,1},** with rules for two binary operators **(+)** and **(.)** and **inversion or complement** as shown In the following operator tables at Figures 1, 2, and 3 respectively.

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

Figure 1

A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

Figure 2

A	A'
0	1
1	0

Figure 3

The following postulates are satisfied for the definition of Boolean algebra on a set of elements S together with two binary operators (+) and (.)

1. (a) Closure with respect to the operator (+).

(b) Closure with respect to the operator (.).

2. (a) An identity element with respect to + is designated by 0 i.e.,

$$X + 0 = 0 + X = X$$

(b) An identity element with respect to . is designated by 1 i.e.,

$$X . 1 = 1 . X = X$$

3. (a) Commutative with respect to (+), i.e., $\quad X + Y = Y + X$

(b) Commutative with respect to (.), i.e., $\quad X . Y = Y . X$

4. The distributive laws of (.) over (+) i.e., **A . (B+C) = (A . B) + (A . C),** and (+) over (.) i.e., **A + (B . C) = (A+B) . (A+C)** can be shown to be applicable with the help of the truth tables considering all the possible values of A, B, and C as under. From the complement table it can be observed that

(a) Operator (.) over (+)

A	B	C	B + C	A. (B + C)	A. B	A. C	(A. B) + (A.C)
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

(b) Operator (+) over (.)

A	B	C	B . C	A+(B . C)	A+B	A+C	(A+B).(A+C)
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

(C) **A + A' = 1**, since, **0 + 0' = 0 + 1 = 1**   **1 + 1' = 1 + 0 = 1**

(D) **A . A' = 0**, since, **0 . 0' = 0 . 1 = 0**   **1 . 1' = 1 . 0 = 0**

5. Postulate 6 also satisfies two-valued Boolean algebra that has two distinct elements **0 and 1** where **0 is not equal to 1**

BASIC PROPERTIES AND THEOREMS OF BOOLEAN ALGEBRA

Principle of Duality

From postulates, it is evident that they are grouped in pairs as part (*a) and (b) and every algebraic expression* deductible from the postulates of Boolean algebra remains valid if **the operators and identity elements are interchanged**.

This means one expression can be obtained from the other in each pair by **interchanging every element** i.e., **every 0 with 1, every 1 with 0**, as well as **interchanging the operators** i.e., **every (+) with (.) and every (.) with (+).** This important property of Boolean algebra is called **principle of duality**.

i.e. **0 <->1 &   + <-> .**

(a) 0 . 1 = 0 <-> (b) 1 + 0 = 1

i.e. (b) is dual of (a)

The following is the complete list of postulates and theorems useful for two-valued Boolean algebra

Postulate 2	(a) $A + 0 = A$	(b) $A.1 = A$
Postulate 5	(a) $A + A' = 1$	(b) $A.A' = 0$
Theorem 1	(a) $A + A = A$	(b) $A.A = A$
Theorem 2	(a) $A + 1 = 1$	(b) $A.0 = 0$
Theorem 3, Involution	$(A')' = A$	
Theorem 3, Commutative	(a) $A + B = B + A$	(b) $A.B = B.A$
Theorem 4, Associative	(a) $A + (B + C) = (A + B) + C$	(b) $A.(B.C) = (A.B).C$
Theorem 4, Distributive	(a) $A(B + C) = A.B + A.C$	(b) $A + B.C = (A + B).(A + C)$
Theorem 5, DeMorgan	(a) $(A + B)' = A'.B'$	(b) $(A.B)' = A' + B'$
Theorem 6, Absorption	(a) $A + A.B = A$	(b) $A.(A + B) = A$

Theorem 1(a): A + A = A

A + A = (A + A).1	by postulate 2(b)
= (A + A) . (A + A′)	by postulate 5(a)
= A + (A.A′)	by postulate 4(b)
= A + 0	by postulate 5(b)
= A	by postulate 2(a)

Theorem 1(b): A . A = A

A . A = (A . A) + 0	by postulate 2(a)
= (A . A) + (A . A′)	by postulate 5(b)
= A (A + A′)	by postulate 4(a)
= A . 1	by postulate 5(b)
= A	by postulate 2(b)

Theorem 2(a): A + 1 = 1

Theorem 2(b): A . 0 = 0   by Duality
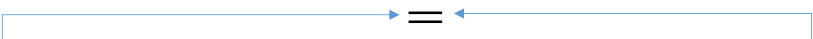
Theorem 6(a): A + A.B = A

A + A.B = A . 1 + A.B   by postulate 2(b)

$\qquad$ = A (1 + B)   by postulate 4(a)

$\qquad$ = A . 1 $\qquad$ by theorem 2(a)

$\qquad$ = A $\qquad\qquad$ by postulate 2(b)

Theorem 6(b): A (A + B) = A   by Duality

Venn Diagram

- **Truth tables :**

➢Both sides of the relation are checked to yield identical results for all possible combinations of variables involved.

➢The following truth table verifies the 6(a) absorption theorem.

X	Y	XY	X+XY
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

➢Following Truth Table verifies the De Morgan's theorem $(x+y)'=x'.y'$

X	Y	X+Y	(X+Y)'	X'	Y'	X'Y'
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

- **BOOLEAN FUNCTIONS**

Binary variables have two values, either 0 or 1.

A **Boolean function** is an expression formed with binary variables, the two binary operators AND and OR, one unary operator NOT, parentheses and equal sign.

The value of a function may be 0 or 1, depending on the values of variables present in the Boolean function or expression. For example, if a Boolean function is expressed algebraically as

$$F = AB'C$$

then the value of F will be 1, when A = 1, B = 0, and C = 1.

For other values of A, B, C the value of F is 0

Boolean functions can also be represented by Truth Tables. A **truth table** is the tabular form of the values of a Boolean function according to the all possible values of its variables. For an n variable, 2^n combinations of 1s and 0s are listed and one column represents function values according to the different combinations of variables

For example, **for three variables** the Boolean **function F = AB + C** truth table can be written as below in Figure

A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

SIMPLIFICATION OF BOOLEAN EXPRESSIONS

- Simplify the Boolean function F = X + (X' . Y)

$$= (X + X') . (X + Y) \quad (4(b))$$
$$= 1 . (X + Y) \quad\quad (5(b))$$
$$= X + Y \quad\quad\quad (2(b))$$

- Simplify the Boolean function F = X . (X'+Y)

$$= (X . X') + (X . Y)$$
$$= 0 + (X . Y)$$
$$= (X . Y)$$

- Simplify the Boolean function F = (X' . Y' . Z) + (X' . Y . Z) + (X . Y')

$$= (X' . Z)(Y' + Y) + (X . Y')$$
$$= (X' .Z) . 1 + (X . Y')$$

Implementation of Boolean functions with gate (Circuit)

Simplify following Boolean Function into minimum number of Literals and Draw Circuit Diagram using AND, OR and NOT Gate

1. F=xy+x'z+yz
2. F=(x+y)(x'+z)(y+z)
3. F=xy+xy'
4. F=xyz+x'y+xyz'
5. F=y(wz'+wz)+xy
6. F=(a + b)' (a' + b')'
7. F = BC + AC' + AB + BCD

Canonical And Standard Forms

Boolean function expressed as a **sum of minterms** or **product of maxterms** are said to be *canonical form*.

Standard form: Sum Of Product (SOP) : AB' + AC' + ABC

Product Of Sum (POS) : (A + B') ▪ (A +C') ▪ (A + B + C)

SUM -> OR gate

PRODUCT -> AND gate

Minterm

A product term containing all N variables of the function in either true or complemented form is called the minterm. Each minterm is obtained by an AND operation of the variables in their true form or complemented form.

For a two-variable function, four different combinations are possible, such as, A'B', A'B, AB', and AB. These product terms are called the fundamental products or standard products or minterms.

Note that, in the minterm, a variable will possess the value 1, if it is in true or uncomplemented form, whereas, it contains the value 0, if it is in complemented form .

Symbol for Minterm is m_j, where j denotes the decimal equivalent of binary number

- For three variables function, eight minterms are possible as listed in the following table in Figure

A	B	C	Minterm
0	0	0	A'B'C'
0	0	1	A'B'C
0	1	0	A'BC'
0	1	1	A'BC
1	0	0	AB'C'
1	0	1	AB'C
1	1	0	ABC'
1	1	1	ABC

Sum of Minterms

$F = A + B'C$

$A = A(B + B')$

$\quad = AB + AB'$

$\quad = AB(C + C') + AB'(C + C')$

$\quad = ABC + ABC' + AB'C + AB'C'$

$B'C = B'C(A + A') = AB'C + A'B'C$

$F = ABC + ABC' + AB'C + AB'C' + AB'C + A'B'C$

$\quad = ABC + ABC' + AB'C + AB'C' + A'B'C$

$\quad = A'B'C + AB'C' + AB'C + ABC' + ABC$

$\quad = m1 + m4 + m5 + m6 + m7$

$F(A,B,C) = \Sigma (1, 4, 5, 6, 7)$

Steps to obtain Sum of MINTERMS

- Find out how many variables is there in equation

- List out all minterms for given variables. i.e m0 to m7 for 3 variables

- Check the given equation is in **SOP FORM** or not? If not then convert it into **SOP form**. i.e **ab + bc + ac'** form

- Find out missing variable in each term & add that variable in that term. i.e for A,B,C: B is missing in AC'. Add B in AC' using AC' = AC'(B+B') = ABC' + AB'C'

- ADD ALL MISSING VARIABLES IN EACH TERMS OF THE EQUATION

- ARRANGE ALL **OBTAINED MINTERMS** IN ASCENDING ORDERS. i.e from m0 to m7 for 3 variables

MAXTERM

A sum term containing all n variables of the function in either true or complemented form is called the maxterm. Each maxterm is obtained by an OR operation of the variables in their true form or complemented form.

Four different combinations are possible for a two-variable function, such as, A' + B', A' + B, A + B', and A + B. These sum terms are called the standard sums or maxterms.

Note that, in the maxterm, a variable will possess the value 0, if it is in true or uncomplemented form, whereas, it contains the value 1, if it is in complemented form.

Symbol for Maxterm is M_j, where j denotes the decimal equivalent of binary number

- Like minterms, for a three-variable function, eight maxterms are also possible as listed in the following table

A	B	C	Maxterm
0	0	0	A + B + C
0	0	1	A + B + C'
0	1	0	A + B' + C
0	1	1	A + B' + C'
1	0	0	A' + B + C
1	0	1	A' + B + C'
1	1	0	A' + B' + C
1	1	1	A' + B' + C'

Product of Maxterms

$F = xy + x'z$

$\quad = (xy + x')(xy + z)$

$\quad = (x + x')\,(y + x')\,(x + z)\,(y + z)$

$\quad = (x' + y)\,(x + z)\,(y + z)$

$x' + y = x' + y + zz' = (x' + y + z)\,(x' + y + z')$

$x + z = x + z + yy' = (x + z + y)\,(x + z + y')$

$y + z = y + z + xx' = (y + z + x)\,(y + z + x')$

$F = (x + y + z)\,(x + y' + z)\,(x' + y + z)\,(x' + y + z')$

$\quad = M_0 M_2 M_4 M_5$

$F\,(x, y, z) = \Pi\,(0, 2, 4, 5)$

Steps to obtain PRODUCT OF MAXTERMS

- Find out how many variables is there in given equation
- List out all maxterms for given variables. i.e M0 to M7 for 3 variables
- Check the given equation is in **POS FORM** or not? If not then convert it into **POS form**. i.e **(a + b)(b + c)(a +c')** form
- Find out missing variable in each term & add that variable in that term. i.e for A,B,C: B is missing in (A + C'). Add B in (A + C') using (A + C') = (A + C')+BB'

  $$= (A + B + C)(A + B' + C)$$

- ADD ALL MISSING VARIABLES IN EACH TERMS OF THE EQUATION
- ARRANGE ALL **OBTAINED MAXTERMS** IN ASCENDING ORDERS. i.e from M0 to M7 for 3 variables

x	y	z	function f1	function f2
0	0	0	0	0
0	0	1	1	0
0	1	0	0	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

F1 = x'y'z + xy'z' + xyz = m1 + m4 + m7

F2 = x'yz + xy'z + xyz' + xyz = m3 + m5 + m6 + m7

- Express the following functions in a sum of minterms and product of maxterms.

  1. $F(A,B,C,D) = D(A' + B) + B'D$

  2. $F(W,X,Y,Z) = Y'Z + WXY' + WXZ' + W'X'Z$

  3. $F(A,B,C) = (A' + B)(B' + C)$

  4. $F(W,X,Y,Z) = (W + Y + Z)(X + Y')(W' + Z)$

Conversion Between Canonical Forms

$F(A, B, C) = \Sigma (1, 4, 5, 6, 7)$ (Sum Of Product) (Sum of Minterms)

$F'(A, B, C) = \Sigma (0, 2, 3) = m0 + m2 + m3$

complement of $F' = F = (m0 + m2 + m3)'$

$$= m0' . m2' . m3'$$

$$= M0.M2.M3$$

$$= \Pi (0, 2, 3) \text{ (Product Of Sum) (Product of Maxterms)}$$

$$mj' = Mj$$

$F (A, B, C) = \Sigma (1, 4, 5, 6, 7) = \Pi (0, 2, 3)$